ERDC/ITL TR-05-4

Information Technology Laboratory

US Army Corps of Engineers®
Engineer Research and Development Center

The USACE DataNet: Providing Efficient Data Access Through a Net-Centric Framework

C. Denise Martin, Cary D. Butler, James T. Stinson, and David C. Stuart

August 2005

ERDC/ITL TR-05-4
August 2005

The USACE DataNet: Providing Efficient Data Access Through a Net-Centric Framework

C. Denise Martin, Cary D. Butler, James T. Stinson, David C. Stuart

Information Technology Laboratory
U.S. Army Engineer Research and Development Center
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

Final report

Approved for public release; distribution is unlimited

Prepared for U.S. Army Corps of Engineers
Washington, DC 20314-1000

ABSTRACT: The U.S. Army Corps of Engineers (USACE) relies on interactive computer-based systems to identify and assess alternatives, make decisions, and solve problems. Data are the principle component of the Science and Engineering (S&E) community that drives the decision making process. Much of the data needed to support the USACE S&E community are available from other Federal agencies. Acquisition of these data is often accomplished via ftp, http, or CD, and results in inefficient and inconsistent use of the data sources. Moreover, data are provided in a myriad of disparate formats and structures, while the models and assessment tools that consume these data require differing formats as well. The efficient handling of data is critical in making appropriate as well as timely decisions. The goal of the DataNet is to streamline the acquisition and dissemination of S&E data across all USACE business areas. This report describes the development and implementation of a common net-centric framework that provides a consistent interface to data sources internal and external to USACE.

Contents

List of Figures

Preface

This report was prepared by Ms. C. Denise Martin, Associate Technical Director, U.S. Army Engineer Research and Development Center (ERDC) Information Technology Laboratory (ITL); Dr. Cary D. Butler, Technical Director, ITL; and Mr. James T. Stinson and Mr. David C. Stuart, Computational Science and Engineering Branch, Engineering and Informatic Systems Division, ITL.

This report was prepared under the general supervision of Drs. Deborah F. Dent, Assistant Director, ITL, and Jeffery P. Holland, Director, ITL.

At the time of publication of this report, COL James R. Rowan, EN, was Commander and Executive Director of ERDC. Dr. James R. Houston was Director.

1 Introduction

1.1 Problem Definition

The U.S. Army Corps of Engineers (USACE) relies on interactive computer-based systems to identify and assess alternatives, make decisions, and solve problems. The principle component of the decision making process is data. Data are defined by Setzer (2001) as a sequence of quantified or quantifiable symbols. Text, numbers, pictures, and animations are all examples of data. Data become information when meaning is applied to them. However, as information is represented as data in a computer, access to data is the focus of this project. For this report, the terms *data* and *information* will be used synonymously.

Data required to support USACE decision-making are available from both internal and external sources. External sources include other Federal agencies (USGS, USDA, NOAA, NASA, etc.) as well as private industry and academia. Acquisition of these data is often accomplished via ftp, http, or CD, and results in inefficient and inconsistent use of the data sources. Moreover, data are provided in a myriad of disparate formats and structures while the models and assessment tools that consume these data require differing formats as well. The efficient handling of data is critical in making appropriate as well as timely, cost-effective decisions. There is clearly a problem when a scientist must spend more time acquiring, manipulating, transforming, and organizing data than analyzing those data.

1.2 Technical Issues

With the introduction of the personal computer in the early 80's, the USACE computing environment, and the computing industry as a whole, changed from a centralized, mainframe environment to a decentralized, desktop environment. As a result, information became scattered over various machines with limited ability to share. Networking technologies were soon deployed to provide a mechanism for sharing information.

During these years of decentralized computing, USACE business components (divisions, districts, laboratories, etc.) have freely addressed their own technology needs with limited emphases on enterprise solutions. Typically, the approach to buying–building, deploying, and maintaining technologies was project-specific, with little or no consideration given to how a technology or its information is used corporately. Not surprisingly, this notion of vertically focusing technology without consideration of the "big picture" is a common problem throughout government and

industry. Today, companies are investing in technology that helps integrate disparate, heterogeneous information and make it available to decision-makers. Industry's acceptance of the web as the information delivery pipeline has sparked the technology industry to develop middleware standards that describe how information is located and shared over the web. The World-Wide Web Consortium (W3C) is the lead for defining Internet-based standards. These web-based technologies address interoperability and security and provide the baseline for all systems, new and old, to work together to improve how technology and information are delivered to customers, business partners, and employees. However, a web-based solution introduces new challenges with respect to security, infrastructure, and management. As applications become dependent on web-based technologies, the computing platforms and communication devices must be able to support the secure, timely delivery of data and functionality. The focus of this project is to exploit these web-based technologies to deliver distributed, heterogeneous information to distributed, heterogeneous applications in a secure and timely manner.

1.3 The Solution—the DataNet

Information is a corporate asset. In fact, it is information that drives our business process, not applications and technology. Applications are developed or purchased to manipulate and create new information. Technology is the enabler that supports applications and the ability to store and deliver information. It is important that all automation efforts focus on information use and not just technology.

The DataNet employs a network-centric approach to streamline and standardize the acquisition, dissemination, and management of data across all USACE business areas.

1.3.1 Objective

The primary objectives of the DataNet are to:

a. Develop, promote, and deploy a common net-centric framework that provides a consistent interface to data sources internal and external to USACE.

b. Formalize connectivity to USACE's data sharing partners (Federal and state governments, universities, industry, and the public).

1.3.2 Report Scope

This report describes the DataNet, USACE's net-centric approach to data acquisition and delivery for Science and Engineering applications. Chapter 2 describes research related to this approach; Chapter 3 describes the framework, or basic components, of the DataNet; Chapter 4 details the implementation of the web components of the framework; Chapter 5 describes some applications that use the DataNet for data acquisition; and Chapter 6 provides a summary of conclusions and describes future challenges.

2 Background

Managing data is expensive and should be performed according to certain basic guiding principles (*Natural Resources Information Mgmt Toolkit, Concise Guide for Technical Managers* 2003) including:

a. Avoid duplication in data acquisition. Share data wherever possible via networks and partnership.

b. Look for existing datasets before collecting data.

c. Adhere to existing government and industry data content, access, and delivery standards.

d. Manage data to maximize their use by multiple processes.

e. Manage data at the owner level and negotiate access arrangements.

f. Require the use of metadata for every dataset.

Service-oriented computing, an innovative approach to computing that uses web services as the building blocks for developing applications (Setzer 2001), provides an effective method for managing data according to these guiding principles.

The Office of Management and Budget (OMB) recently established the Federal Enterprise Architecture Program Management Office (FEA-PMO) to prepare a roadmap to support the implementation of the President's priority E-Government initiatives (*Charter and Operating Principles, Solution Architect Working Group (SAWG)* 2002). The roadmap outlines a component-based architecture that defines a set of recommendations that should be considered when selecting tools, technologies, and standards for business solutions. The component-based architecture provides the basis for interoperability, sharing, and reuse among government systems. The benefit of this architecture is a reduction in the total cost of ownership, shorter software development and testing cycle, consistency, and the ability to manage intra-agency functions, data, and technology more effectively. In June 2003, the FEA-PMO published the *Service Component Reference Model (SRM)* Version 1.0 as a foundation for government-wide improvement. According to the SRM, the effective identification, assembly, and usage of components allows for aggregate services to be shared across agencies. Service component aggregation enables rapid building of components to support a given initiative. The SRM is one of five reference models within the overarching Federal Enterprise Architecture (FEA). The FEA *Technical Reference Model (TRM)* (2003) describes the technology that supports the implementation of component-based architectures

defined in the SRM. The FEA-TRM recommends the use of web services as access channels for service components. The *Corps Enterprise Architecture–Technical Reference Model (CeA-TRM)* (CeA Project Delivery Team 2003) establishes a Common Computing Environment (CCE) to provide technical direction for software and hardware components that require interfacing at the USACE enterprise level. One of the characteristics of the CCE is a net-centric environment in which computer networks facilitate the connectivities among USACE applications and data. The CeA-TRM recommends the use of web services to deliver data and functionality.

Industry leaders, such as Microsoft, Sun, and IBM ESRI are also embracing a service-oriented architecture. According to Heather Kreger of IBM, "web service technologies are being developed as the foundation of a new generation of business to business (B2B) and enterprise application integration (EAI) architectures (Kreger 2003)." John Williams of Sun Microsystems contends that web services "have the potential to dramatically mitigate the complexities and the costs of integration projects (Williams 2003)." Further, Williams points out that the competition between Sun's J2EE specification and Microsoft's .NET framework for web service implementations should not detract from the core commitments both companies have to making web services successful as common communications infrastructure.

Web services provide a collection of technical standards and communication protocols that use the Internet to facilitate programmatic access. Web Services are based on the following W3C standards:

2.1 XML

The eXtensible Markup Language is designed to improve the functionality of the Web by providing more flexible and adaptable information identification. It is called extensible because it is not a fixed format like HTML (a single, predefined markup language). Instead, XML is actually a *metalanguage*—a language for describing other languages—that lets you design your own customized markup languages for limitless types of documents (Flynn 2003).

2.2 SOAP

The Simple Object Access Protocol uses a combination of eXtensible Markup Language (XML)-based data structuring and the Hyper Text Transfer Protocol (http) to define a standardized method for invoking methods in objects distributed in diverse operating environments across the Internet. Client applications make remote procedure calls to SOAP "services," which are basically code libraries/objects with exposed methods. According to the W3C specification, SOAP is a lightweight protocol for exchange of information in a decentralized, distributed environment. It is an XML-based protocol that consists of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined datatypes,

and a convention for representing remote procedure calls and responses (Hadley et al. 2003).

2.3 WSDL

Web Services Description Language (WSDL) is a specification for describing web services based on XML. A WSDL file contains all of the information needed to interact with a SOAP service, such as input parameters, type, and number for method input, as well as the output parameters, type, and number for method output. It also contains the URL address of the SOAP service, and the SOAP encoding scheme that is used. The WSDL file serves as a contract between the client application and a service provider. If a service provider publishes a WSDL file for a specific service, and the WSDL is not valid for use with the said service, then the provider is not meeting the obligations of this contract (Curbera et al. 2001).

2.4 UDDI

The Universal Description, Discovery, and Integration (UDDI) is a specification for distributed web-based information registries of Web Services. UDDI registries are used to promote and discover distributed web services. Designed to assist software developers in finding available services, it contains all of the information necessary to describe a service, how it is used, and where it is located (Bellwood et al. 2002).

The purpose of a Web service is to programmatically expose a process, or function, over a network through an open, standardized communication mechanism and format. Whereas web applications are designed for browsers, web services are designed for applications. Client applications consume web services via a request/response 'call.' In a service-oriented architecture, as shown in Figure 2.1, the service provider develops a web service and associated WSDL service description for use by other client applications, or service requestors. The service provider publishes the service description in a UDDI registry so that others can locate the service. The client application finds the service via the UDDI registry and uses the WSDL description to interact with the service through the service provider (Ferris and Farrell 2003).

Commercial vendors are offering integrated development environments (IDE) that hide the details of SOAP/XML encoding/decoding and transmission to the remote web service. Toolkits, such as Microsoft's Visual Studio .NET, generate the WSDL and install all the programs on a web server. Client applications access the web service just as they would access any other object class. The toolkit reads the WSDL, translates the class definitions into one of many programming languages (C#, Java, C++, VB, etc.), and builds the proxy and stub code necessary to communicate between the client and the remote web service (Barclay et al. 2002).

Two principal architectures are used for web service interfaces: synchronous web services and asynchronous web services. These two architectures are

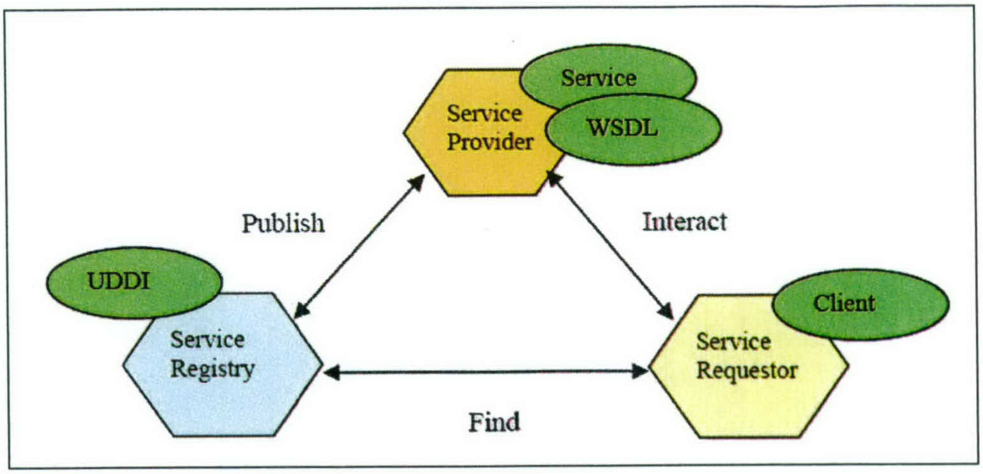

Figure 2.1. Service-Oriented Architecture

distinguished by their request-response handling. With synchronous services, client applications send a request to a service and then suspend their processing while they wait for a response. With asynchronous services, client applications initiate a request to a service and then resume their processing without waiting for a response. The service processes the request and returns a response at some later point. Deciding which architecture is best depends on the types of work the service performs and the available technologies. Synchronous services are best when the service can process the request in a small amount of time and when applications require an immediate response to a request. When a web service requires complex processing that may require minutes or hours to complete, an asynchronous architecture is desirable so that the client application can continue with some other processing rather than wait for the response (Sun Microsystems 2002).

Although asynchronous behavior is not explicitly supported by SOAP, the IDE toolkits provide an asynchronous interface through the use of a begin method to invoke the service, an end method to poll for the invocation status and fetch the result, and a callback method that provides the status check without polling (Barclay et al. 2002).

In summary, web services use standard web-based protocols that can traverse firewalls in a cross-platform environment to allow differing systems to interoperate. Web services are characterized by their reusable modularity, their availability for use by distributed software systems, their machine-readable description, their implementation independence, and their publishability via service repositories (Fremantle et al. 2002). According to Francisco Curbera, "the web services framework intends to provide a standards-based realization of the service-oriented computing paradigm, which has emerged in response to a fundamental shift in the way enterprises conduct their business. Fully integrated enterprises are being replaced by business networks in which each participant provides the others with specialized services. Traditional IT infrastructures in which infrastructure and applications were managed and owned by one enterprise are giving way to networks of applications owned and managed by many business partners. Standards and the pervasiveness of network technologies provide the technology support for this trend (Curbera et al. 2003)."

3 A Framework for the DataNet

The DataNet provides a one-stop-shop for data acquisition from within USACE, from other Federal and state agencies (USGS, NASA, EPA), and from industry (ESRI, Microsoft). Science and Engineering (S&E) client applications, such as computational models, web applications, GIS applications, and portable devices, can access heterogeneous data via the DataNet. In order to define the DataNet, it is helpful to describe the layers that compose the overall data framework, as shown in the diagram in Figure 3.1.

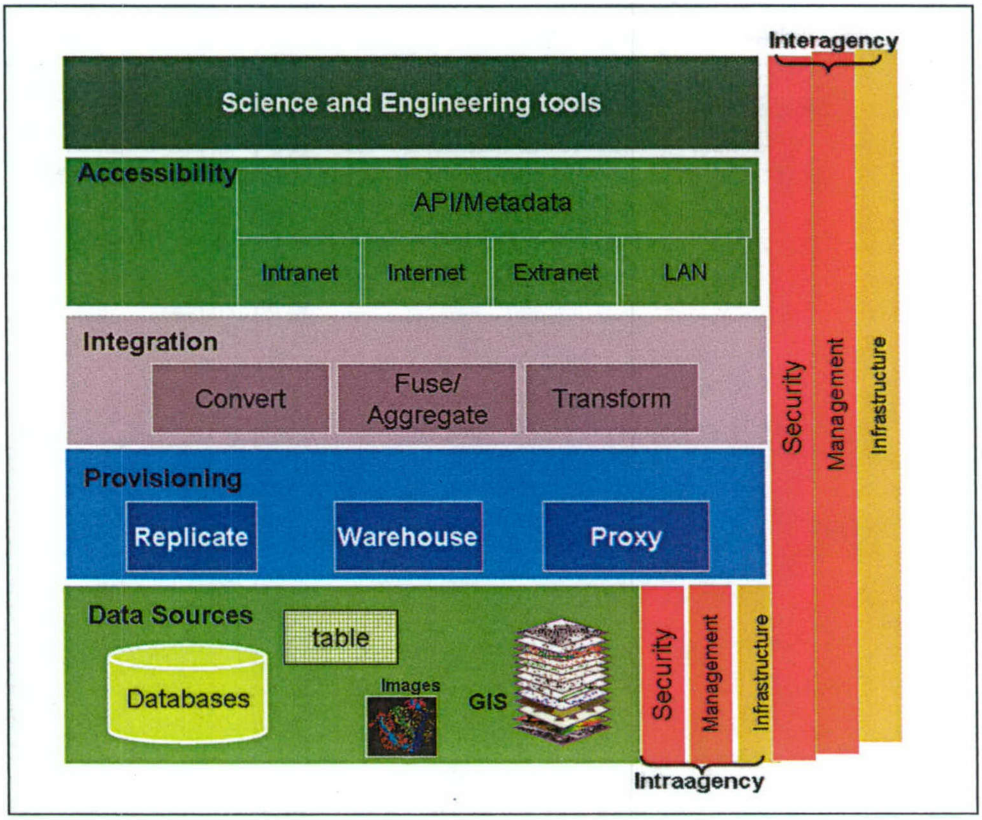

Figure 3.1. DataNet framework

The top level of the framework is the set of S&E applications that require access to the data. These applications range from simple desktop screening level

tools to commercial GIS software operating on a shared server, to multidimensional models operating in a supercomputing environment. The challenge is to develop a framework that will support data accessibility by all of these applications. The remainder of this chapter provides an overview of the layers that make up the DataNet framework.

3.1 Data Sources

At the base of the framework is the Data Source layer, which includes the basic 'raw' data that S&E applications require, such as an Oracle database, an Excel table, a binary file, or an image file. These sources are stored and maintained in varying formats on distributed servers within many different organizations and are governed by intra-agency security, management, and infrastructure policies and constraints within their native environments.

The typical scenario for locating and accessing these data involves website downloads, ftp site downloads, or CD exchanges for each user of each application. Additionally, each user stores the data locally and preprocesses it for use by each application. The overhead associated with this process is enormous. Jacquez et al. (2001) estimated that scientists spend 80 percent of their effort locating data and reading them into software applications, 15 percent of their effort preprocessing the data, and only 5 percent of their time actually performing modeling and analysis. USACE estimates are similar, with 70 percent of our effort spent locating data and reading them into applications, 15 percent spent pre-processing the data, and 15 percent spent performing modeling and analysis. Clearly, a significant time savings could be achieved through a corporate approach to data connectivity, i.e., the DataNet.

3.2 Provisioning

At the Provisioning level, individual data sources are prepared for delivery to distributed S&E applications. The DataNet supports three approaches to provisioning: replicating the data source on an in-house server, warehousing specific data sources on an in-house server, and providing a proxy mechanism for direct delivery of the data from the source.

3.2.1 Replication

Replication involves the physical copying of data from one data source to another. Frequently, required external data are available for user access via downloadable files from websites or ftp sites or CDROM. To provide programmatic access to the data, copies of the data files are stored on an in-house server. The NOAA estuarine bathymetry data are an example of this type of provisioning. NOAA provides access to the data via downloadable files from the NOAA web site, *http://spo.nos.noaa.gov/bathy*. All of the files were downloaded to a central server and standard access mechanisms were developed to deliver the data to applications.

3.2.2 Data Warehouse

A Data Warehouse is an enterprise-wide repository that replicates data from publication tables on different servers and platforms to a single subscription table. This effectively consolidates data from multiple sources. Data are extracted from heterogeneous sources, translated to required formats, and the resulting data are loaded into tables within the data warehouse (Stanford University 2003). Automated data staging tools facilitate the data extract, manage data transformation, data merging, and aggregation. The USACE CorpsMap database is an example of a warehouse approach to data provisioning. The CorpsMap geospatial database, which resides on a USACE Central Processing Center server, includes a comprehensive nation-wide base map consisting of numerous data layers, such as GDT Dynamap, USGS National Map, USACE Navigation Data Center Data, and many others.

Both warehousing and replicating data sources require a plan for periodic updates of the data source, as well as software and hardware maintenance. The primary advantage of both the data warehouse and data replication approaches is that USACE is not dependent on other agencies' data access strategies. The main disadvantage is that USACE incurs the cost of maintaining copies of other agencies' data or duplicate copies of USACE data sources.

3.2.3 Proxy

The proxy approach introduces a proxy component that acts as an intermediary between S&E applications and data sources. The proxy effectively hides the details of the data location, encoding schemes, and communication protocols from the client application. Web services (explained in detail in Chapter 2) will be used to implement the proxy approach. Web services provide a collection of technical standards and communication protocols that use the Internet to facilitate programmatic access. A web service provides a single point of programmatic access to data sources for use by multiple applications. Today's applications are typically built on technologies and protocols intended for human (user) consumption, not system (programmatic) consumption. According to Christopher Koch, "The web services vision is to enable computer systems and business processes to seek each other out over the Internet, lonely hearts style, and have deep, meaningful interactions with no human intervention" (Koch 2003).

The promise that web services will efficiently facilitate interoperability has led many companies and organizations to invest in web service projects at the enterprise level. Caution is in order, however, because web services are such a new technology and industry standards are not fully developed. Oellerman (2001) contends that the success of web services corresponds directly to the extent of our ability to agree on what web services are and how they are implemented. If we define them differently, it will be inevitably difficult to build and consume them across various implementations. The DataNet framework defines web service implementation guidelines for USACE. It is important to note that, although a web service may be developed and maintained by USACE, the data it delivers are stored and maintained by the agency who owns the data. In cases where a web service delivers data from external sources, Service Level

Agreements (SLA) must be established with other agencies to ensure the availability, stability, and performance of the data services within specified constraints. Chapter 4 provides additional information describing the implementation of web services.

3.3 Integration

Data sources vary significantly in format, structure, and content; therefore, some level of preprocessing is needed to properly adapt the data for their most effective use. The Integration layer provides mechanisms for tailoring data to meet the needs of specific applications, such as data aggregation or fusion services, coordinate conversion services, subsetting services, or format conversion services.

3.4 Accessibility

The Accessibility layer defines the network gateways for the DataNet and the interface and metadata information necessary for application developers to access data sources via the DataNet. Data sources are connected to the DataNet by publicly accessible Internet gateways, a publicly accessible but restricted Extranet gateway, an internally accessible Intranet gateway, as well as local area networks. Application Programming Interfaces (API) provide a set of routines, protocols, and tools that application developers use to access DataNet data. Thus, one consistent set of data access tools is developed and provided to application developers to access specific data sources. Metadata registries then provide information, such as which network segment to access, required security mechanisms, example code for accessing specific data sources, and technical points of contact.

3.5 Overarching layers

From an interagency perspective, three overarching concerns span every layer: security (information assurance), management, and infrastructure.

3.5.1 Security

Security issues pervade every layer of the DataNet. The security measures imposed on the DataNet must be able to interoperate with the varying levels of security associated with individual data sources, especially external sources. If we think of the DataNet as a collection of nodes that represent common access to data, with links between those nodes representing network connections, the primary security issues deal with controlling access to the various nodes.

a. *Network Gateways*. One method to control access is through the selection of network gateways (Internet, Extranet, Intranet, LAN). As the DataNet operates on a collection of network servers homed to one of two Internet gateways—the Corps of Engineers Enterprise Infrastructure Services

(CEEIS) Internet gateway or the Defense Research and Engineering Network (DREN) gateway—security measures are well defined for those gateways. Security devices, including gateway router, stateful firewall, VPN concentrator, intrusion detection devices, site intrusion detection devices, and site firewalls, are monitored 24 hours a day, 7 days a week. Access to the USACE computer resources is limited to users who have a valid requirement, through the use of hardened passwords and permissions. Information Assurance Vulnerability Alerts (IAVA's) are monitored by HQ USACE and Department of the Army for strict compliance. To filter hostile traffic, virus packages from Antigen, Norton, and McAfee are used. Routine hardware and software upgrades, backups, and monitoring of usage metrics are provided.

b. *Encryption*. A second required security measure for web applications involves the use of Secure Sockets Layer (SSL) encryption. The use of SSL encryption ensures that all traffic, including user-ids and passwords, is encrypted as it passes between the client application and the server.

c. *Authentication*. Thirdly, all applications that interface with the DataNet are required to go through an authentication process. This is the first line of defense to manage access to the DataNet, as well as to control the use of computational and networking resources. Authentication is the process of assuring that someone is who they say they are. An authentication web service was developed to provide a standard method for controlling access to specific components of the DataNet. The service authenticates on the basis of a set of authentication sources, which are managed sets of user ids and passwords. Currently, authentication sources include the Corps User-ID and Password System (UPASS) and the Army Knowledge Online (AKO) user-id and password system. Once users (a user can be a person or an application) are authenticated, it is important to define which DataNet components are available to them. Access rights are defined through the use of user communities and profiles.

As the Department of Defense continues to issue Common Access Cards to DoD employees, the DataNet will extend the authentication sources to include the CAC. This will increase security by ensuring that the user actually has in his or her possession a DoD issued Common Access Card and associated digital certificates.

3.5.2 Management

The Management layer encompasses those activities that control the maintenance of components within the framework, as well as processes associated with the framework, such as standards, service level agreements, change control, and monitoring of components. A network-based framework for data delivery demands a managed process to ensure quality of service. We must be prepared to manage the assimilation of ever-changing technology into our business process. Standards, which govern data content and format as well as data transfer protocols, provide the basis for storing and delivering data from disparate sources.

a. *Service Level Agreements*. A service level agreement is a formal contract between a service provider and a service consumer that guarantees quantifiable network performance at defined levels (Myerson 2002). The contract outlines key performance measures, such as service availability, server response time, service repair time, service technical support, within which the service provider agrees to operate and deliver its services. An SLA should also specify exceptions in terms of failures, network issues outside the control of the service provider, denial of service, and scheduled maintenance. An example of an SLA is provided in Appendix A With respect to the DataNet, it is critical that SLAs are developed for web services that access external data sources.

b. *Monitoring*. Monitoring provides the capability to track various metrics about each DataNet web service, such as:

(1) Is the service operational?

(2) Who is using the service?

(3) When is the service most often used?

(4) How long does it take for the service to complete a request?

(5) How much data are being sent to and returned from the service?

This information is valuable for security and maintenance reasons and provides the quantification necessary to monitor SLAs. Usage monitoring functionality is provided as a web service and must be referenced as an object in all DataNet services.

3.5.3 Infrastructure

The Infrastructure layer includes the physical computing and networking resources required to make the DataNet a reality. For internal data sources and replicated external data sources, the Corps of Engineers Enterprise Infrastructure Services (CEEIS) provides the Corps' primary information technology infrastructure asset. This asset consists of world-class corporate data processing and global networking, enabling the Corps' programs and business processes. Functionality and capability are provided in a manner that remains robust, viable, and meets customer expectations while maintaining a secure and cost-conscientious culture. As the Corps' information systems and network communications infrastructure provider, CEEIS facilitates Corps-wide data administration and secured information exchange and provides the necessary worldwide automation and communications environment for the development, deployment, operation, and maintenance of corporate resources.

Infrastructure support for external proxied data sources is provided through the establishment of a corporate Web Farm. The Web Farm provides the Internet, Intranet, and Extranet gateways that support the Accessibility layer of the DataNet framework. The Web Farm maintains the computing platforms, software, networks, and security necessary to support the deployment of web services within the DataNet framework.

4 A Web-Centric Implementation

While the DataNet framework encompasses all networks, including LANs, the emphasis of this report is on the web-based aspect of the DataNet. This chapter will describe the web portion of the DataNet framework.

4.1 Technical Approach

Service-Oriented Computing (SOC) provides an efficient architecture for managing data according to the guiding principles listed in Chapter 1 (Papazoglou and Georgakopoulos 2003). The web-centric components of the DataNet are based on the SOC paradigm that employs web services as the basic components for developing applications. Using this SOC paradigm, we identified data sources that support the USACE decision-making processes and were connected to the DataNet via the development of web services. Each web service was developed on the basis of the World Wide Web Consortium (W3C) standard technologies Simple Object Access Protocol (SOAP) and Web Services Description Language (WSDL) and published to a registry that describes the service and how to use it. Each web service operates in a secure computing environment on the USACE Corporate Web Farm. Service Level Agreements are being established for all services outside the USACE network space. The remainder of this report will describe this approach with respect to the DataNet framework described in Chapter 3.

4.2 Data Sources

S&E applications often share common data requirements, although in inconsistent formats. Examples of data requirements include financial data, environmental data, hydrologic data, meteorological data, topographic data, infrastructure data, property data, etc. To more efficiently identify and manage these data requirements, data flow among applications, and data access across multiple business areas, it is helpful to develop an information architecture. The architecture provides a formal blueprint that enables an organization to develop standard methods for organizing, storing, processing, analyzing, and accessing common sources and types of data. For example, an information architecture recently developed for the USACE Regional Sediment Management program

identified the following sources and types of data required to support many of the software applications used to perform regional sediment management (Table 4.1).

Table 4.1
Data Sources for RSM

Category	Data Source
Elevation	USGS National Elevation Data
	NOAA Estuarine Bathymetry
Weather	University of Utah Meso West Precip/Weather
	METAR current surface conditions
	NCDC Precipitation
Hydro	USGS Stream Flow
	Corps Water Management System Realtime Gage data
Infrastructure	USACE National Inventory of Dams
Land Use/Vegetation	USGS Land Use/Land Cover Data
Soils	USDA STATSGO
Maps/Imagery	Various sources

These data are stored and maintained by many different agencies, in many different formats, on many different platforms. Currently, every user of every application must locate and access these data via website downloads, ftp site downloads, or CD exchanges. Additionally, each user stores the data locally and pre-processes it for use by each application. The overhead associated with this process is enormous. A more efficient approach is to allow data owners to store and maintain the data in their native environment, while developing web-based mechanisms for controlled access.

4.3 Provisioning

Based on the data requirements elaborated in the Information Architecture, connectivity to the data listed in Table 4.1 was established using a service-oriented, or proxy, approach. The web service technology and the details of this implementation are described in the following paragraphs.

Web services were developed or procured for each of the data sources listed in Table 4.1. For most of the data sources, a web service was developed that directly connects to the data in their native environment. This includes data stored in relational databases, binary files, ASCII text files, geodatabases, etc., and includes both internally managed data sources and externally managed data sources. For example, the USACE National Inventory of Dams is an Oracle database that resides on a USACE Central Processing Center server. Using a Web service to share database information gives the added capability of being able to connect from anywhere on the Web. Suppose a client application requires information from the NID shared database (Figure 4.1). The application sends a

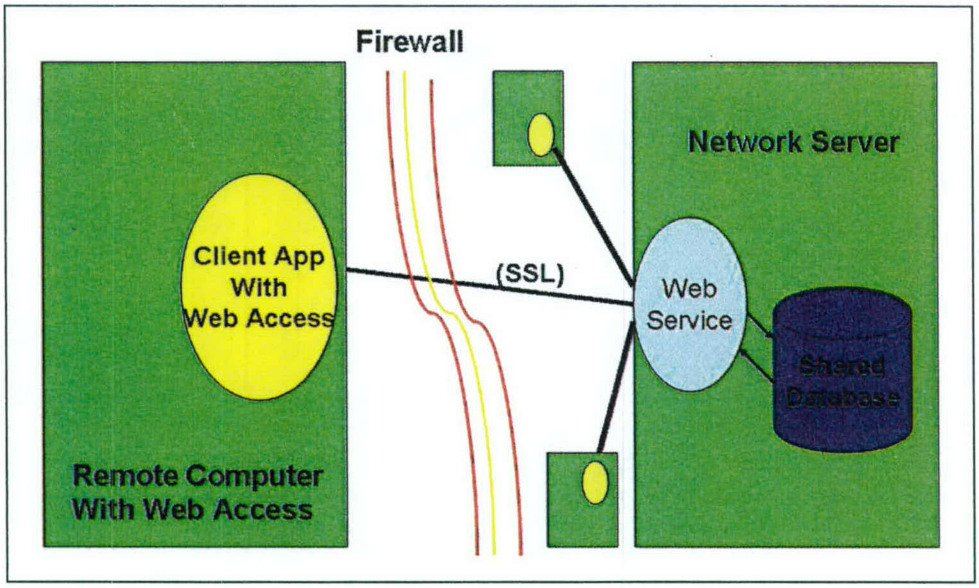

Figure 4.1. Database service

request for information to the Web service in the form of an SQL query or some other parameter string, along with a security token. If the security token is valid, the Web service forwards the request to the database. The resulting data are returned to the Web service, converted to XML and returned to the client application.

The USGS National Elevation Dataset (NED) is accessed somewhat differently by issuing requests to the Web application managed by USGS for elevation data within a given geographic area. It is important to note that, although this web service is developed and maintained by USACE, the data are stored and maintained by USGS. USACE is currently in the process of establishing service level agreements with USGS to ensure the availability, stability, and performance of the data sources.

In some cases, direct access to the data source was not feasible. Consequently, it was necessary to replicate a copy of the data source on a USACE server and develop a web service to connect to the USACE copy of the data source. The NOAA Estuarine Bathymetry data service is an example of this approach. Access to these data is provided via downloadable files from the NOAA web site, ***http://spo.nos.noaa.gov/bathy***. To programmatically access the data, all of the files were downloaded and stored on a Web Farm server. A DataNet service was developed to access those files based on a given geographic area. This approach requires a plan for periodic updates of the data source, as well as software and hardware maintenance. The primary advantage of this approach is that USACE is not dependent on NOAA's data access strategies; however, USACE incurs the cost of maintaining copies of their data. To provide programmatic access to various sources of maps and images, the decision was made to establish a contract with a commercial vendor, ESRI, to purchase access to their web services. With the contract in place, a user-id and password are provided by ESRI. The web service call requires a valid user-id and password to

request access to the set of services. The service returns a valid key that is used to access the WSDL files, published in a service registry. The WSDL provides a machine-readable description of the service and is used to identify the interface of the service, its location, and access information. Multiple USACE applications can programmatically access multiple sources of imagery and maps (with no feature data) without the burden of storing and maintaining these large data sets.

4.4 Integration

Web services provide an efficient technology for the integration layer of the DataNet, because the code can be developed once and invoked by many different applications. While the web services described in the previous section access data in their native format, it is usually beneficial to deliver the data in some standard format. Therefore, many of the services include code that converts the data before delivering them to an application. Many of the S&E applications require access to more than one data service. Obviously, each application can synchronously call each data service, but it is more efficient to develop an aggregation service that acts as a proxy service for locating and consuming other DataNet services. The data aggregation service calls the DataNet services in parallel, asynchronously, which allows client applications to poll the service for status updates as data are acquired.

4.5 Accessibility

All of the web services that compose the DataNet are currently operating on the USACE Web Farm via Extranet gateways currently restricted to USACE and Army users. A *Service Registry,* based on the UDDI specification, provides the mechanism by which software developers find available services and information describing their use. Controlled access (currently only USACE employees) to the Service Registry is available at *https://cdf.usace.army.mil*. Services within the registry are categorized according to service types, such as data services, utility services, etc., to assist users in locating specific services. The Registry provides the following information for each web service:

Name	Service
Service Description	Brief description that explains what the service provides
WSDL URL	URL location of the WSDL file that describes this service
Service Info/Help URL	Overview—homepage with specific information about this service (example: Descriptions of input and outputs to service methods, package information and explanations, example XML SOAP requests, etc.)
Client Code Download	Example client code that calls the service, or a small code library that can help end users call the service
Category	Category that best describes the type of service, i.e., data service, utility service, image service, etc.
City	City in which the service will be physically located (management purposes)
State	State in which the service will be physically located (management purposes)
Zip Code	Zip code in which the service will be physically located (management purposes)

All of the web services that currently compose the DataNet have been registered in the Service Registry, categorized as Data Services. The categories listed in Table 4.1 were used as sub-categories in the taxonomy to facilitate searching for various types of data services, i.e., elevation, weather, hydro, infrastructure, etc.

4.6 Overarching Layers

Because the DataNet operates on the USACE Corporate Web Farm, the security architecture and infrastructure for the Web Farm applies to the DataNet.

4.6.1 Security

The DataNet exploits the following specific security features:

a. *UserId's/Passwords*—all passwords used on web farm servers will conform with the Corps User-ID and Password System (UPASS) standards or the Army Knowledge Online (AKO) standards, or both; independent passwords are issued for ORACLE access to selected databases; access to DataNet services uses independent passwords issued for CEEIS web access or AKO user-id and passwords.

b. *User profiles*—users are restricted to specific DataNet service access to accomplish their specific tasks,

c. *Views*—views are used to segregate data and services access, permitting users to access only the data or services necessary to accomplish their tasks.

d. *Encryption*—the use of Secure Sockets Layer (SSL) encryption ensures that all traffic, including user-ids and passwords, is encrypted as it passes between the user's web browser and the server; all DataNet services use SSL protocols.

e. *Auditing*—some applications make extensive use of how and when given services are accessed, as well as how data definitions and data manipulation are executed.

f. *Software*—all operating system software, web server software, and application server software are maintained at current, supported versions and all vendor and Army recommended patches and service packs are applied.

g. *Backup and Continuity of Operations*—all data stored on web farm servers are backed up to magnetic tape in both incremental and regular full backups. Scheduled maintenance on web farm machines, which requires downtime, is scheduled in advance, outside of normal duty hours. For any extended periods of web site outage, http requests can be redirected to another server that gives users an expected availability time for the data that they are seeking. Web farm servers are covered by

hardware manufacturers' warranty plans, which permit rapid resolution and correction of hardware related outages;

h. *Scanning*—all web farm servers are regularly scanned for IA vulnerabilities by the IAT and any vulnerabilities detected are promptly corrected.

The DataNet services are currently in the process of being certified and accredited under the Defense Information Technology Security Certification and Accreditation Process (DITSCAP), which implements DoD Instruction 5200.40 *DoD Information Technology Security Certification and Accreditation Process.*

4.6.2 Infrastructure

The DataNet currently operates on four Dell 1650 Pentium 4 servers on the Web Farm. Service usage is monitored to provide an indication of infrastructure requirements. As usage increases, the infrastructure will be upgraded to support it.

4.6.3 Management

All DataNet web services were developed according to the W3C standards described in the Provisioning section in this chapter.

a. *Configuration Management.* Software configuration management (SCM) for the services is assisted by the use of the comprehensive SCM software, Perforce, which features a scalable client–server architecture. Requiring only TCP/IP, Perforce supports version control, workspace management, atomic change transactions, and a powerful branching model to develop and maintain multiple versions of code.

b. *Service Level Agreements.* Web services have led to more challenging and complex SLAs that guarantee a certain quality of service, encompassing availability, reliability, and response time, to ensure uninterrupted business operations. USACE is currently involved in developing SLAs with USGS and NOAA to ensure the reliable availability of the data sources to which our web services connect.

c. *Operations and Maintenance.* A DataNet Administration Team (A-Team) will be formed to manage the day-to-day operations and maintenance of the Service Registry and the technical documentation associated with registered services. The A-Team will include designated web farm team members, testing team members, and technical advisors, and will be tasked with the following responsibilities:

    (1) Administering databases for access management, and registry metadata and updates.

    (2) Testing of proposed new services.

    (3) Updating of existing services.

(4) Coordinating with service providers.

(5) Providing technical assistance in service development.

(6) Upgrading hardware and software.

(7) Backing up software.

(8) Monitoring and analyzing usage metrics.

d. *Testing*. A draft Testing Plan has been developed to describe the basic functional requirements of all DataNet web services, as well as a set of procedures for testing the services' operation. As new services are developed, the registration process is managed as follows:

(1) The service provider (developer) must provide to the Registry Administrator a copy of the service, required Registry information, technical documentation describing the service, and a client application that consumes the service.

(2) The A-Team will conduct tests to determine network impacts, code reliability, and security, as applicable.

(3) The results of the tests will be provided to the owner of the service and any corrections must be made by the owner.

(4) The corrected or modified service is resubmitted to the A-Team, retested, and, if accepted, it is registered in the Service Registry by the Registry Administrator. Any further modifications to the service must be approved by the owner and coordinated with the Registry Administrator.

e. *Technical Transfer*. The DataNet is transferred to users in the following ways: 1) short (1–2 hour) seminars provide a basic overview of the DataNet; 2) workshops (1–2 days) include the basic overview, technical details, demonstrations, and user feedback; 3) technical guidance documentation describes how to develop and consume DataNet services, set up a web service development environment, etc.; 4) a web portal provides the mechanism for organizing technical documentation, presentations, meeting minutes, related articles, DataNet services, and reusable applications and code libraries; 5) a Steering Group provides a conduit for influencing development, thus ensuring that it supports the needs of users and that users understand how it supports them; 6) the A-Team manages the day-to-day operations and maintenance of the Service Registry, the technical documentation associated with registered services, as well as technical assistance in service and application development.

5 Applying the DataNet

The DataNet provides a standards-based, cross-platform web-centric framework that allows software developers the capability to use heterogeneous operating systems and development environments. Several client applications have already embraced the DataNet as a source for data acquisition and delivery, including a desktop browsing application, an extension to a commercial software product, and a legacy hydrological modeling system. Each of these applications consumes the web services connected to the DataNet to support some of its data requirements. The purpose of this chapter is to describe how each application is utilizing the DataNet.

5.1 S&E Data Browser

The S&E Data Browser, which serves as a common gateway interface to access data needed for S&E modeling, is a thin-client stand-alone .NET application programmed in C# using Visual Studio .NET software. Most of the functionality executes on remote servers, responding to web service calls. A user selects from a list of S&E data sources the data he or she requires and a geographic area of interest (Figure 5.1). The location of data sources available for the selected area is displayed. The user then selects the features for download and requests the data. The data aggregation service, described in the Integration section of Chapter 4, acts as a proxy service for locating and consuming DataNet services for the selected data sources. This configuration makes the Data Browser application easier to maintain as new data services are added to the DataNet. A time limit is imposed for responses from each DataNet service. Acquired data are stored in a .zip file for download. The Data Browser also uses commercial ArcWeb services to display background maps, such as topographic maps, aerial photographs, satellite images, and road atlas maps.

5.2 ArcGIS Extension

The ArcGIS extension increases the functionality of ArcGIS to include display and download of data sources available via the DataNet (Figure 5.2). This software is also a .NET application programmed in C# using Visual Studio .NET, but it depends on the ArcGIS software. Basically, the user selects a data source

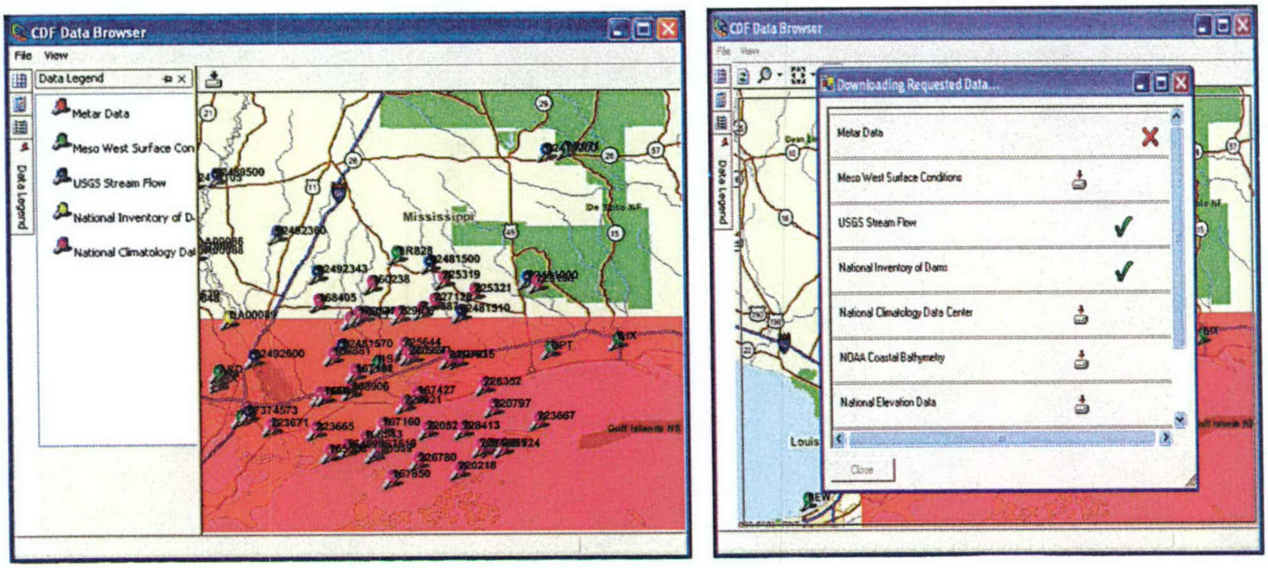

Figure 5.1. S&E Data Browser

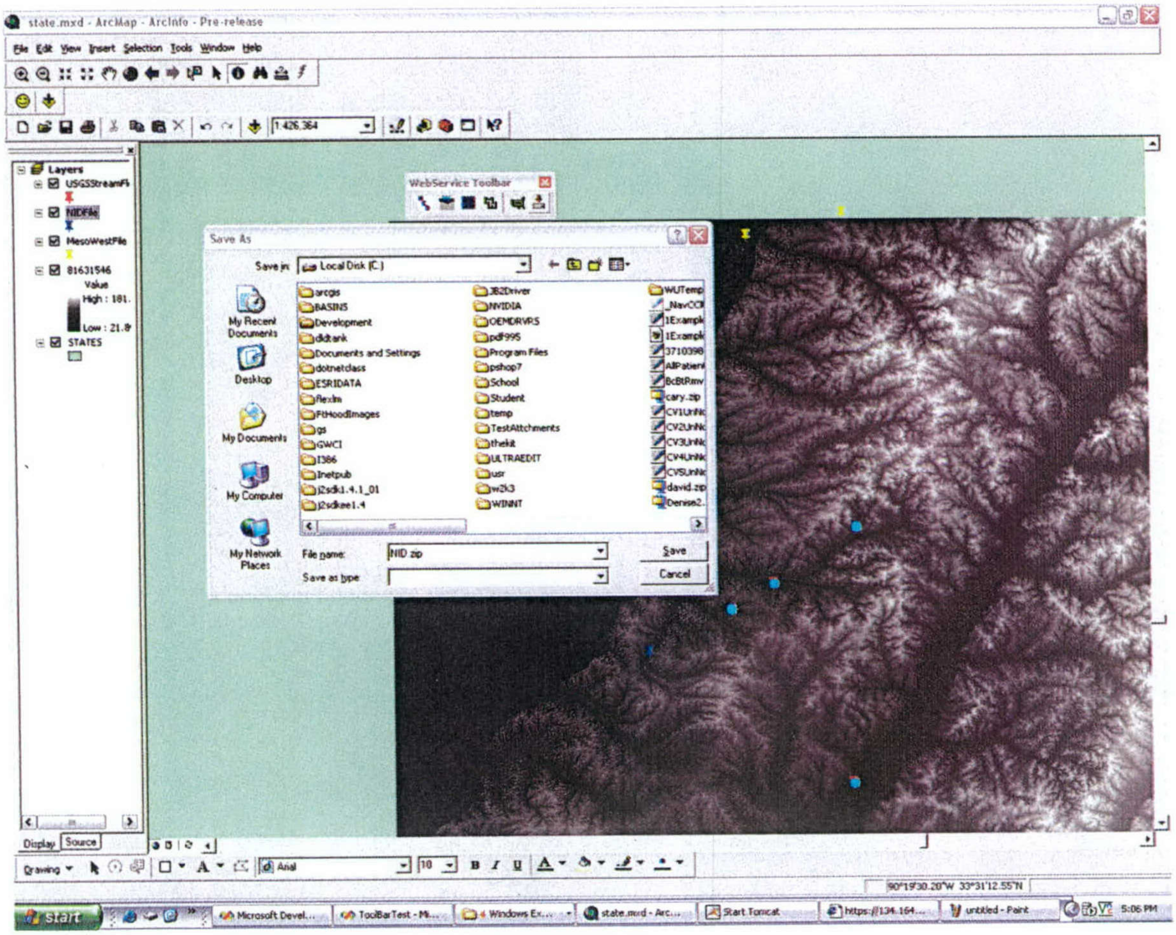

Figure 5.2. ArcGIS Extension using DataNet Web Services

that he or she desires to include in an ArcGIS analysis and a geographic area of interest. The application requests the data from the appropriate DataNet service, retrieves the data, creates an ArcGIS data layer, displays the layer, and allows the user to download the data as a .zip file (raster data) or XML (point data) file.

5.3 Legacy Application Enhancement

The Watershed Modeling System (WMS) provides a comprehensive environment for hydrological analysis of watershed systems. Developed by USACE, WMS provides graphical tools for use in the delineation of watersheds, as well as an interface to multiple hydrologic computational models. It serves as a preprocessor of data used in watershed delineation and as input to models. WMS is a Win32 application based on Microsoft Foundation Classes (MFC). Legacy applications present a unique challenge to changing technology. For example, it is not possible to call .NET SOAP services in MFC applications because MFC libraries do not support SOAP. However, MFC applications can communicate with Microsoft COM libraries and COM libraries can communicate with .NET SOAP services. Therefore, a Microsoft COM library was developed that calls the selected DataNet services in much the same way that the ArcGIS extension does.

These applications represent three very different environments that require access to the same data sources. In all three applications, the time required to acquire and format model-ready data for a 100- × 100-km area of interest was minutes rather than hours. Because access to these data sources was available as a standard web service via the DataNet, the software developers for all three applications were able to provide increased functionality, programmatically, that drastically reduced the time that users previously spent locating, acquiring, and managing data.

6 Conclusions

6.1 Summary

In defining a network-centric approach to data acquisition, dissemination, and management, USACE has indeed accomplished its goal of streamlining the acquisition and dissemination of S&E data across all USACE business areas. The DataNet is consistent with the basic guiding principles (*Natural Resources Information Mgmt Toolkit, Concise Guide for Technical Managers* 2003) of data management by providing a solution that avoids duplication in data acquisition, facilitates the sharing of data, both internal and external, via networks and partnerships, adheres to standards, promotes owner-level management and service level agreements, requires metadata for data and services, and is accessible by distributed, heterogeneous applications,.

We have shown that: 1) the DataNet provides a one-stop-shop for data acquisition from within USACE, from other Federal and state agencies (USGS, NASA, EPA), and from industry (ESRI, Microsoft); 2) Science and Engineering client applications, such as computational models, web applications, GIS applications, and portable devices, can access heterogeneous data sources via the DataNet in a timely and secure manner. The DataNet framework truly provides a standards-based realization of the service-oriented computing paradigm.

6.2 Future Challenges

The work that has been accomplished to date and described in this report provides a sound basis for net-centric data acquisition, dissemination, and management. However, there remain at least two primary challenges: 1) establishing a network of "trusted" external data sources, and 2) changing the data sharing culture within the USACE S&E community.

As Federal agencies are beginning to embrace the service-oriented computing paradigm, the issue of trusted data sources has already surfaced. At a Federal Land Management Agencies Geospatial Architecture Conference in November 2002, the main action item at the close of the meeting was to form an Intra-agency Working Group to build a Land Management Web Services Repository. Although this action item has not yet been accomplished, this is the type of future activities that will be needed to take the net-centric approach to the next level.

Change is typically met with resistance initially. When the approaches described in this report were presented as concepts 2 years ago, many of the USACE Scientists and Engineers were skeptical at best. Over the past 9 months, those attitudes have slowly begun to change as demonstrations of the capabilities became available. Transferring this technology to the USACE scientists and engineers who develop software, or who are responsible for the development of software through contracts, remains a challenge. A formal Technology Delivery Plan will be developed later this year by a heterogeneous group of USACE scientists and engineers, with the expectation that this group will claim ownership of this approach and spread it among their peers.

References

Barclay, T., Gray, J., Strand, E., Ekblad, S., and Richter, J. (2002). "TerraService.NET: An Introduction to Web Services," Technical Report MS-TR-2002-53, Microsoft Research Advanced Technology Division, Redmond, WA, June.

Bellwood, T., et al. (2002). "UDDI Version 3.0, UDDI Spec Technical Committee Specification," OASIS, July.

CeA Project Delivery Team (2003). "Corps Enterprise Architecture—Technical Reference Model," version 1.0, September.

Curbera, F., Meredith, G., and Weerawarana, S. (2001). "Web Services Description Language (WSDL) 1.1 W3C Note," *http://www.w3.org/TR/wsdl*, 15 March.

Curbera, F., Khalaf, R., Mukhi, N., Tai, S., and Weerawarana, S. (2003). "The Next Step in Web Services," *Communications of the ACM* 46(10), 29–34.

Federal Enterprise Architecture Program Management Office (2002). "Charter and Operating Principles, Solution Architect Working Group (SAWG)," version 1.0, draft, April.

Federal Enterprise Architecture Program Management Office (2003). "The Service Component Reference Model (SRM)," version 1.0, June.

Federal Enterprise Architecture Program Management Office (2003). "The Technical Reference Model (TRM)," version 1.1, August.

Ferris, C., and Farrell, J., (2003). "What are Web Services?" *Communication of the ACM* 46(6), 31.

Flynn, P. (2003). "The XML FAQ, v. 3.01," *http://www.ucc.ie/xml/#acro*, 14 January.

Fremantle, P., Weerawarana, S., and Khalaf, R. (2002). "Enterprise Services," *Communications of the ACM* 45(10), 77–82.

Hadley, M., Mendelsohn, N., Moreau, J., Nielsen, H. F., and Gudgin, M. (2003). "SOAP Version 1.2 Part 1: Messaging Framework," *http://www.w3.org/TR* , 24 June.

Jacquez, G. M., Greiling, D., and Kaufman, A. (2001). *Spatial Pattern Recognition in the Environmental and Health Sciences: A Perspective*, BioMedware and TerraSeer, Inc., Ann Arbor, MI.

Koch, C. (2003)."The Battle for Web Services," *CIO*, 1 Oct., pp. 54-64.

Kreger, H. (2003). "Fulfilling the Web Services Promise," *Communications of the ACM* 46(6), 29–34.

Myerson, J. (2002). "Guarantee your Web Service with an SLA," *http://www-106.ibm.com/developerworks/webservices/library/ws-sla/*, 1 April.

"Natural Resources Information Mgmt Toolkit, Concise Guide for Technical Managers," (2003). *http://www.nlwra.gov.au/archive/toolkit/concise_guide/concise_guide.html*, 7 November.

Oellermann, W. L. (2001). *Architecting Web Services*, Springer-Verlag, New York.

Papazoglou, M. P., and Georgakopoulos, D. (2003). "Service-Oriented Computing," *Communications of the ACM* 46(10), 25–28.

Setzer, V. W. (2001). "Data, Information, Knowledge, and Competency," *http://www.ime.usp.br/~vwsetzer/data-info.html*, 4 April.

Stanford University (2003). *What is a Data Warehouse?*, *http://www.datawarehousing.com.*

Sun Microsystems (2002). "Using Web Services Effectively," *http://java.sun.com/blueprints/webservices/using/webservbp3.html.*

Williams, J. (2003). "The Web Services Debate," *Communications of the ACM* 46(6), 59–63.

Appendix A
Pixxures WebPix Service Level Agreement

The Pixxures' WebPix Service Level Agreement outlines key performance measures within which Pixxures endeavors to operate and deliver its WebPix Services. Each measure is composed of an indicator that can be quantified, a related standard of performance, and a specific Service Level Target. Service Level Targets and automatic Service Level Credits become effective 30 days following the In Service Date.

Pixxures is responsible for network operation and availability, facility infrastructure, equipment, security, and web-deployed applications, depending upon the terms of the specific agreement with the Customer.

Key Performance Measures

Measure	Indicator	Standard	Service Level Target[1]
Service Availability	Time during which the web service is available for use	7 days by 24 hours	99.7% averaged over a monthly period
Service Performance Measures	Web Server Response Time[2] to an http request for up to 100,000 pixel image	Less than 5 seconds	99.7% of performance log entries reporting <5 second response, averaged over a monthly period
	Web Server Response Time[2] to an http request for up to 1,000,000 pixel image	Less than 8 seconds	99.7% of performance log entries reporting <8 second response, averaged over a monthly period
Service Repair Time	The duration required to repair service from the time a service outage is detected or reported	Less than 4 hours	99.7% average of all repairs over a monthly period
Service Support	Pixxures provides a 24X7 Help line which provides an escalated notification to Pixxures = technical Support Staff	Less than 30 minutes from receipt of notification to response to Customer	100%

[1] Service Level Target does not include time spent during routine and scheduled maintenance windows. Pixxures will inform company in advance of scheduled maintenance windows.
[2] Server Response Time is defined as the elapsed time between when the request is received at Pixxures' Web server and the time the image is ready for sending. Pixxures is not responsible for delays associated with available Internet bandwidth, Internet routing, or Customer-based communication limitations such as firewalls, proxy managers, or internal networks.

Facility Maintenance

	Mountain Standard Time
Routine/Scheduled Maintenance	Sundays between 01:00 and 06:00 Tuesdays between 02:00 and 06:00
Advance Notice of Routine/Scheduled Maintenance	At least 5 business days
Non-scheduled maintenance	Negotiated as required

Service Level Limitations

Pixxures bears no responsibility for any decline in Service Levels that is attributable to the Customer, the Customer's agents or subcontractors or a Customer Supplied Component including any Customer supplied or modified scripts.

Pixxures is not responsible for web services supplied by third party providers that are linked to the Pixxures WebPix services. Should the failure of any of these web services cause the WebPix service to drop below Service Level Targets, Pixxures will use best efforts to resume services from the third party or engage a replacement provider.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* August 2005	**2. REPORT TYPE** Final report	**3. DATES COVERED** *(From - To)*
4. TITLE AND SUBTITLE The USACE DataNet: Providing Efficient Data Access Through a Net-Centric Framework		**5a. CONTRACT NUMBER**
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) C. Denise Martin, Cary D. Butler, James T. Stinson, David C. Stuart		**5d. PROJECT NUMBER**
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Engineer Research and Development Center Information Technology Laboratory 3909 Halls Ferry Road Vicksburg, MS 39180-6199		**8. PERFORMING ORGANIZATION REPORT NUMBER** ERDC/ITL TR-05-4
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Corps of Engineers Washington, DC 20314-1000		**10. SPONSOR/MONITOR'S ACRONYM(S)**
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The U.S. Army Corps of Engineers (USACE) relies on interactive computer-based systems to identify and assess alternatives, make decisions, and solve problems. Data are the principle component of the Science and Engineering (S&E) community that drives the decision making process. Much of the data needed to support the USACE S&E community are available from other Federal agencies. Acquisition of these data is often accomplished via ftp, http, or CD, and results in inefficient and inconsistent use of the data sources. Moreover, data are provided in a myriad of disparate formats and structures, while the models and assessment tools that consume these data require differing formats as well. The efficient handling of data is critical in making appropriate as well as timely decisions. The goal of the DataNet is to streamline the acquisition and dissemination of S&E data across all USACE business areas. This report describes the development and implementation of a common net-centric framework that provides a consistent interface to data sources internal and external to USACE.

15. SUBJECT TERMS

Data acquisition	Data sharing	Service-oriented computing
Data management	Net-centric framework	Web services
	Service-oriented architecture	

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(include area code)*
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR	36	